

REMARKS

Introduction

In view of the foregoing amendments and following remarks responsive to the Final Office Action dated November 18, 2008, Applicants respectfully request favorable reconsideration of this Application.

Applicants respectfully thank the Examiner for conducting a telephonic interview with Applicants' undersigned representative on January 12, 2008, during which the parties discussed the final rejection in this case.

Since this reply is being filed in response to a Final Office Action, Applicant will focus comments on the Response to Arguments section of the Final Office Action.

Claim 1

With respect to claim 1, Applicant has previously argued that the workflow diagram of Bailey does not teach drawing a line between objects to represent assignment to or definition within another object of the OOP. In rejecting this argument, the Office stated:

"Bailey teaches joining symbols representing objects to define a functionality that interrelates the objects (see Fig. 4A-D) and there is clear object-oriented factory-based building wherein object classes with their associated members, pre-defined properties and methods are fetched/instantiated as from an available object factory in Bailey's visual tool (see col. 7, line 53-66; col. 4, lines 10-28), which renders the argument about a mere "work-flow diagram" misplaced. There is nothing compelling as requirement in the language (emphasis added)

recited as "line between each object....and another object in the....graphical representation *to which it is assigned or within which it is defined*", so that such language clearly precludes the linking of graphical icons by Bailey—drawing of lines to denote their functional dependency (see Fig. 4D, 8D) in terms of their point of junction or nature thereof (e.g. *terminals...sink object...source subject* – col. 16, lines 5-31) – from reading onto this "representation to which it is assigned or within which it is defined," which appears to be a broad relationship relating to a "graphical representation." Iconic representations joined one another by a link for depicting association with another iconic presentation in an OOP signifies designer-based (a) visual assignment of an entity to a graphical representation of other object/entity, (b) definition of an entity within the scope of another graphical representation. Thus, objects drawn with lines to join them (see Fig 4, 8) is a visual tool like in Bailey's OOP approach using factory of predefined objects class and methods entail a development stage wherein any hand-linking by the developer is for assigning a symbol property or iconic functionality to another symbol representing some object(s) initially created by the designer in the visual enlistment palette/pane as shown in the rejection. Specifically, the very act of joining two symbols with specific bubble to denote some form of terminals suffice to meet "to which it is assigned or within which it is defined", with "it" being either a symbol, a icon, or a class object."

After careful review of the above quoted text from the Final Office Action and the discussion with the Examiner during the telephonic interview, it appears that it is the Office's position that the grammar in step 5 of claim 1, and particularly, the use of the indefinite pronoun "it" leaves the claim open to different interpretations. More particularly, it appears to be the Office's position that it is unclear if the "it" referred to in step 5 refers to a symbol, an object, or an icon, and to which symbol, object, or icon. Applicants have herein rewritten step

5 of claim 1 to eliminate the use of indefinite pronouns and to clarify that the things being discussed are the objects corresponding to the symbols drawn in the diagrams.

As such, the claim can no longer be read broadly so as to encompass any line between two objects in Bailey, as the Office seems to have asserted.

Accordingly, claim 1 should patentably distinguish over the proposed combination of Bailey and Kodosky for the reasons set forth in response to the previous Office Action (the claims now being more clearly commensurate with the previous argument).

Particularly, the Office asserts that Bailey teaches in column 4, lines 23-26, step (5) of claim 1 of drawing a line between each object drawn in step (4) and another object in the graphical representation to which it is assigned or within which it is defined.

This is not accurate. Rather, column 4, lines 23-26 of Bailey state:

According to the invention, these symbols can be linked together by the developer in the form of a data flow or block diagram that logically represents the flow of data and control information into, out of, and between the selected control objects.

This section of Bailey discloses nothing but a flow diagram, which shows the order of steps, or a block diagram, which shows the flow of data between blocks.

This has essentially nothing to do with step (5) of claim 1. Specifically, step (5) of claim 1 does not recite simply drawing a line between objects, but drawing a line between objects to represent two very specific concepts in OOP, namely, assignment to or definition within another object in object oriented programming.

The present invention provides a technique for graphically illustrating this concept in an overall system and method for representing object-oriented programming concepts. Column 4, lines 23-26 of Bailey does not relate to this concept, let alone disclose the specific features claimed in step (5) of claim 1. This concept is completely lacking in Bailey.

Claim 3

With respect to dependent claim 3, which depends from claim 1 and recites the additional step of graphically denoting the main object in the diagram by drawing another symbol around the symbol for the main object, Applicants had previously argued that the Office's reliance on Kodosky for teaching graphically denoting the main object and Visio as teaching drawing another symbol around the symbol for the main object was misplaced. Particularly, Applicant had argued that Visio discloses nothing but a drawing program that can be used to draw a circle around another graphical representation, but does not suggest doing so for any particular reason.

In response to this argument, the Office responded:

"The 'denoting' of a main object is a necessary step in Bailey, because in a framework design where a base object is instantiated by user selection whereby additional objects are linked or added to associate with this main object, this starting step is taught in Bailey otherwise the process of expanding the start object with additional members, objects (i.e. as cited in Bailey: col. 10 lines 36-40, 53-55) would not have been able to go further. Double-clicking or highlighting this object would have been analogized to 'denoting', and manually effectuating a focus-type of

event (i.e. user actions) in visual design (such as Bailey) where mouse click/movement is preponderant suggest circling the selected or highlighted item. A peripheral line of circular or rectangular shape drawn at the contour of one object of interest (see Kodosky) to enable it to be more visible for further action has been further exemplified in contours in Visio. That it, highlighting an icon by way of a mouse click (see Bailey: col. 13 middle) would have been another of such means to draw a contour, whether this is circular or rectangular, for easy focus as explained above. There is sufficient teaching in Bailey's manual selecting and enlisting of objects on a visual frame, so that combined with the hierarchy of objects (as in Kodosky's having a base object atop), and the contours set forth in Visio, the 'denoting' limitation in terms of a drawing circular contour (around a base object) would have been an obvious limitation. The Applicants fail to point out—via factual evidences to the contrary—how using a manual clicking of mouse in Bailey in light of a base hierarchical object in Kodosky combined with the highlighting (square contour) in Visio, fail to teach the 'denoting' as recited, such that the combination as proffered would have been non-obvious. The argument is not persuasive."

Applicant respectfully traverses. First, this argument is confusing because it seems inconsistent with the actual rejection of claim 3, which relies on Kodosky paragraph [0012], not Bailey, for teaching denoting a main object.

In any event, the Office's assertion that "the 'denoting' of a main object is a necessary step in Bailey, because in a framework design where a base object is instantiated by user selection whereby additional objects are linked or added to associate with this main object, the starting step is taught in Bailey otherwise the process of expanding start object with additional members, objects (i.e. as cited in Bailey: col. 10, lines 36-40, 53-55) would not have been able to go further" is

improper. There is absolutely no basis for the Office to conclude that Bailey requires the main object to be denoted in any special way. The Office appears to be asserting that, because other objects need to be linked or associated with this main object in Bailey that the main object must be denoted as the main object. However, there is nothing in Bailey or the knowledge of the ordinarily skilled artisan that compels this conclusion. The only thing that is necessary in Bailey in order to "go further" is to have an object represented on the screen. The Office fails to indicate any reason why it would be necessary to denote the main object in Bailey, let alone by drawing another symbol around it. In fact, there is not even a "main object" in Bailey and the Office has essentially conceded so (by virtue of relying on Kodosky for teaching a main object). Accordingly, the Office's rationale is faulty.

There is simply no reason to conclude that, in Bailey, it would be impossible to "go further" without "denoting" the "main object."

Furthermore, the Office asserts --

Clicking or highlighting this object would have been analogized to 'denoting' and manually effectuating a focus-type of event (i.e. user actions) and individual design (such as in Bailey) where a mouse click/movement is preponderance suggests circling the selected or highlighted item. A peripheral line or circular or rectangular shape drawn at the contour of one object of interest (see Kodosky) to enable it to be more visible for further action has been further exemplified in contours by Visio. That is, highlighting an icon by way of a mouse click (see Bailey: col. 13 middle) would have been another of such means to draw a contour, whether this is circular or rectangular, for easy focus as explained above.

From this quote, it appears to be the Office's position (1) that Bailey (which does not even have a "main object") teaches clicking on an object, (2) that it is well known that clicking on an object often causes that object to be highlighted, (3) that highlighting an object is like denoting it by drawing another symbol around it, and (4) that it would have been obvious to combine all of these features to result in the step recited in claim 3 of denoting the main object by drawing another symbol around it.

This is nothing but a piece meal reconstruction of the present invention using the present specification as a road map. There are a multitude of errors in the above-noted analysis of the Office. First, there is no main object in Bailey. Second, the Office appears to believe that it is in the prior art that "a peripheral line or circular or rectangular shape drawing of the contour of one object of interest (see Kodosky) to enable it to be more visible for further action has been further exemplified in contours by Visio." This has many problems in and of itself. First of all, a teaching (and it is unclear whether the Office believes this teaching appears in Kodosky or Visio or somewhere else) of a peripheral line "to enable it to be more visible for further action" is irrelevant to claim 3. Claim 3 does not recite the step of denoting of an object to make it more visible. It recites denoting an object as the main object. The alleged teaching of a peripheral line to make an object more visible simply is not commensurate with the claim scope and is largely irrelevant in the absence of prior art suggesting doing so in connection with a main object of the diagram. The encircling of the main object in claim 3 is

not to make it more visible. Rather, it is to denote it as the main object. This is not in the prior art.

Second, it appears to be the Office's position that clicking on an object highlights it. However, the statement assumes facts that cannot be assumed. First, clicking on an object only highlights the object if the particular program that is being used is designed to do so. The Office is assuming that mouse clicking causes highlighting in Bailey. But there is no basis for this conclusion.

Second, even if Bailey did teach this, as soon as the user clicked on another object, the first object would be unhighlighted and the second object would be highlighted. This can hardly be considered to be a teaching of "graphically denoting the main object." Rather, this would be nothing more than a teaching of graphically denoting the object that the mouse happens to be positioned over when it is clicked.

There is no support for the conclusion that the prior art of record teaches what is claimed in claim 3.

Claim 7

Claim 7 recites that the method of claim 1 is used to document preexisting object oriented programming software.

In Section D of the Response to Arguments section of the Final Office Action, the Office rejected this argument asserting "these allegations are referred back to the responses regarding 'drawing a line' among symbols and the use of

base object to start enlisting further object oriented as set forth in sections B and C.”

Although Applicant is not sure it fully understands this rejection, the reference to sections B and C of the Response to Arguments section seems to indicate a continuation of the faulty logic that the Office appears to be applying throughout this Office Action. Particularly, the Office seems to be asserting that the fact that the prior art provides the means to execute the recited steps is sufficient to suggest the actual step. In other words, the Office seems to be saying that one could use the proposed combination of Bailey and Kodosky to document software. However, this is simply insufficient. There has to be a suggestion of the recited step (i.e., documenting preexisting object oriented programming software), not merely the tools to do so.

On the other hand, it appears that the Office may not be interpreting claim 7 the way Applicants had intended. The present invention can be used to specify or create software. On the other hand, as claimed in claim 7, it can also be used to document preexisting software. This is the embodiment that claim 7 is reciting. Applicants have herein amended claim 7 to make more explicit that the software to be documented is preexisting software. As Applicant explained in the response to the previous Office Action, the Office appears to be relying on Kodosky for this teaching, but the cited portion of Kodosky states that his engine is used for specifying or creating distributed systems and/or applications. Kodosky does not talk about documenting existing software. Thus, the fact that

Kodosky teaches other embodiments, wherein it is used for creating software is not relevant to claim 7.

Claim 10

With respect to claim 10, which recites that the second diagram does not disclose objects assigned to and defined within the first object and the first diagram does disclose objects assigned to and defined within the first object, the Office appears to have accepted Applicant's previous arguments because it has asserted new grounds of rejection that no longer relies on Paragraph 15 of Kodosky with respect to claim 10. Particularly, the Office now asserts:

Based on the hierarchy of OO objects in Kodosky and the well-known concept that base class includes members defined therein in set forth in Baileys, the above submember of a base class would have flow out of the OOP approach in both Bailey (refer to claim 1 and Kodosky). Enabling a base class to disclose its defined member as opposed to the other way around (e.g. the second object not disclosing base/first object it is assigned to) would have been a obvious feature adopted in Bailey (in light of Kodosky).

The new rejection of claim 10 does not appear to address the scope of the claim. In short, the Office's position essentially is that, because base classes include members defined therein, it is obvious to represent that in one diagram and not in another diagram. The Office's statement that enabling a base class to disclose its defined member as opposed to the other way around is obvious simply does not address what is recited in claim 10. Assuming for the sake of argument that this statement is correct and supported by the prior art (both of

which Applicants dispute), none of this would lead the skilled artisan to decide to show the relationship in the first diagram and not show it the second diagram, as claimed. In the present invention as claimed in the claims on which claim 10 depends, i.e. claims 1 and 9, there is no reason that the assignments and definitions cannot be shown in both diagrams, or only the second diagram (as defined in the claim). However, claim 10 defines a specific paradigm for doing so that Applicants deem particularly useful, namely, showing it only in the first diagram (which is defined in the claims as the diagram in which the first object is the main object).

Accordingly, claim 10 potentially distinguishes over the prior art of record.

Claims 13, 14, and 16

With respect to the claims 13, 14, and 16, which each recites a plurality of different types of objects that are denoted by different types of symbols, it is now clear from the interview, that it is the Examiner's position that, contrary to the Applicants' intent, these claims did not require a different symbol for each of the different objects recited in the claim. That is, for instance, referring to claim 13, it recites five different symbols and states what type of OOP object each different symbol represents. Based on the interview, it is now clear that it is the Office's position that the term "include" in the claim does not have the meaning that Applicant intended, namely, that this claim requires five different symbols respectively representing the five different types of objects recited. Rather, the

Office believes that this claim reads on Bailey because Bailey discloses some of these symbol object types.

While Applicant respectfully disagrees with the Examiner's reading of the term "include", the Examiner did suggest that changing the claim language to read "include at least the five following symbols" would cure the problem. Applicant has no problem doing so, and has amended claims 13, 14, and 16 accordingly. Hence, claims 13, 14, and 16 clearly distinguish over prior art of record insofar as, by the Office's own admissions, the prior art of record discloses only some of the symbol types.

Claim 19

With respect to claim 19, which recites that the symbols representing method type objects recited in claim 13 are drawn connected to the main object of the diagram and represent that the object is available within that main object and does not represent that the main object invokes it, the Office has asserted a new grounds of rejection as compared to the previous Office Action. Particularly, in the previous Office Action, the Office relied on Bailey, col. 4, lines 35-40 and col. 7, lines 57-60 and col. 8, lines 14-18 as teaching this feature. In the latest Office Action, the Office instead relies on Kodosky, Figs. 35 and 28A and paragraph 314, as teaching this feature. Particularly, the Office asserts that "Kodosky teaches objects with line to point to the underlying methods (Stop 846, Shut-Down 850-Fig. 35, 28A) and main program symbols drawn connected to subprograms to execute functions described to be enclosed for the base object

(para 0314)" and asserts that it would have been obvious to combine this with Bailey.

Applicant respectfully traverses the new rejection. Applicant also respectfully points out that this is a new rejection of claim 19, which has not been amended (nor any of the claims from which it depends). Thus, it is a new grounds of rejection that was not necessitated by any amendments made by Applicants. Accordingly, this last Office Action cannot be a Final Office Action in accordance with MPEP §706.07(a), which states "Under present practice, second or any subsequent actions on the merits shall be final, except where the examiner introduces a new ground of rejection that is neither necessitated by applicant's amendment of the claims, nor based on information submitted in an information disclosure statement".

Thus, as an initial matter, Applicant respectfully requests the Office to withdraw the finality of the November 18, 2008 Office Action as being incompatible with MPEP §706.07(a).

Furthermore, Applicants respectfully traverse the rejection in any event. Particularly, claim 19 recites that the method type object symbol is drawn connected to the main object and represents that the object is available within that main object and "does not represent that the main object invokes it." The Office asserts that "Kodosky teaches objects with line to point to the underlying method... (citations omitted) and main program symbols drawn connected to subprograms to execute functions described to be enclosed for the base object (para 0314)." Thus, the Office is asserting that Kodosky teaches essentially the

opposite of what is claimed. Claim 19 recites that the connection between the two objects does not indicate that the main object invokes it, whereas the Office is asserting that Kodosky teaches that the connection between the two objects denotes that the one object executes the function described by the other object. These are, virtually, opposites. At the very least, they are inapposite.

Thus, even accepting the Office's position, Kodosky teaches something different than what is claimed in claim 19. Accordingly, the rejection is insufficient.

Even further, however, cited paragraph 314 of Kodosky relates to neither of Figs. 35 or 28A or symbols 846 or 850 referred to by the Office.

Even further, paragraphs 402 and 403 of Kodosky, which do relate to the symbols 846 and 850 and Fig. 28A do not discuss the lines between symbols, but comprises what appears to be an entirely irrelevant discussion of the various icons in Figures 28A and 28B.

Accordingly, claim 19 patentably distinguishes over the prior art insofar as the prior art does not teach that to which has been cited and, even if it did, it is the opposite of what is claimed.

Conclusion

In summary, in the above amendments and remarks, Applicants have attempted to amend claims 1, 7, 13, 14, and 16 to address the alleged shortcomings in the claim language so that Applicants' previous arguments as to how these claims distinguish over the prior art are now commensurate with the

claim language in the Examiner's view. Furthermore, Applicant has pointed out how the rejection of claim 3 is deficient under the law. Finally, Applicant has addressed the new rejections of claims 10 and 19, showing how they are legally deficient and also constitute new grounds of rejection, therefore, making it inappropriate to make the last Office Action final.

Accordingly, independent claim 1 patentably distinguishes over the prior art for all the reasons set forth by Applicant in response to the previous Office Action. All of its dependent claims, claims 2-21, therefore, distinguish over the prior art for at least all of the same reasons. Even further, however, dependent claims 3, 7, 10, 13, 14, 16, and 19 even further distinguishes over the prior art for the reasons set forth herein above and in the previous response.

Finally, independent claim 22 distinguish over the prior art for the reasons set forth in the response to the previous Office Action and hereinabove in connection with claims 1, 3, 22, 23 and 24. Particularly, Applicant has herein amended independent claim 22 to now correspond substantially to the subject matter of claim 3, including the limitations of its base claim, claim 1. More particularly, the limitations of former claims 23 and 24 have been added into independent claim 22 (and claims 23 and 24 have been cancelled). Accordingly, claim 22 recites all of the distinguishing features discussed above in connection with claims 1 and 3 and distinguishes over the prior art for the same reasons discussed above in connection with claims 1 and 3.

Claims 24-35, which depend from claim 23, distinguish over the prior art of record for at least all of the reasons the set forth above in connection with claim

22 as well as all the reasons previously asserted in response to the previous Office Action.

In view of the foregoing remarks, this application is now in condition for allowance. Applicants respectfully request the Office to issue a Notice of Allowance at the earliest possible date. The Examiner is invited to contact Applicants' undersigned counsel by telephone call in order to further the prosecution of this case in any way.

Respectfully submitted,

Dated: January 16, 2009

/Theodore Naccarella/

Theodore Naccarella, Reg. No. 33,023
Saul Ewing LLP
Centre Square West
1500 Market Street
Philadelphia, PA 19102

Tele: (215) 972-7877

Fax: (215) 972-4161

Attorneys for Applicants